

AD-A064 235

ROYAL SIGNALS AND RADAR ESTABLISHMENT MALVERN (ENGLAND)
PLANNING AND CONTROL OF SOFTWARE BASED SYSTEMS, (U)

F/G 9/2

MAY 78 K JACKSON

UNCLASSIFIED

RSRE-78012

DRIC-BR-65548

NL

| OF |
ADA
064235



END
DATE
FILMED
DDC

4 - 79

LEVEL II

2

Procurement Executive, Ministry of Defence
Royal Signals and Radar Establishment

Report No 78012

Planning and Control of Software Based Systems

by

K Jackson

SUMMARY

This paper embodies the report of a committee of The Defence Scientific Advisory Council which studied the problems associated with the development and maintenance of software based computer systems. The report identifies the nature of the problems in this area, makes some comments on the problems and puts forward eight recommendations which it is suggested could alleviate many of the problems.

-A-

May 1978

ADDRESS LINE	
STN	WHITE BRIER
SCC	SOY BRIER
CLASSIFIED	
JOURNALISATION	
BY	
DISTRIBUTION/AVAILABILITY FORM	
REG	AVAIL. BY/OF SPECIAL
A	

Copyright
C
Controller HMSO London
1978

DDC
REF ID: A
FEB 7 1979
RECEIVED
D

1 INTRODUCTION

This report is essentially the same as a report produced by a committee of the Defence Scientific Advisory Council of which the author was secretary at the time. This version has been produced with the objective of making the information more widely available.

2 TERMS OF REFERENCE

A working party was set up with the following terms of reference:

- (a) To study the problems associated with the development and maintenance of computing systems in the software field in co-operation with RSKE.
- (b) To report on measures which need to be taken to improve planning and control.
- (c) In particular to consider whether any tools based upon computing facilities could usefully be developed so that they would be applicable to more than one project.

A number of discussions were held with project teams selected on the basis of advice given by the chief scientists of the three services. It was not the purpose of the working party to comment on any particular project specifically, but rather on the effects of methods of procurement in general.

3 NATURE OF THE PROBLEM

The working party studied the three major problems common to most software-based projects, namely:

- (a) It is sufficiently common to be a matter for concern that software-based projects arrive late and cost more than was budgeted.
- (b) The projects commonly require more hardware and much more software than was originally planned.
- (c) At delivery the systems are commonly found to be in a form which is not convenient for use. Much post-delivery work is then needed to convert them into the form that the users require. It is sometimes found that in the course of this work the users can trim down the system so that it becomes more economic than that first delivered. It should be noted that software development, unlike hardware development, is in practice performed partly by users.

4 COMMENTS ON THE PROBLEM

The Inter Establishment Committee on Computer applications have produced a "Guide to the Development of Computer Based Systems", issued by the Procurement Executive of the Ministry of Defence (IECCA (P) 4/72 (1)), Part 1 of this discusses the management of Computer Based Systems in terms of the need to proceed by constructing a series of 'prototypes', each of which serves to define the project more closely than the last. The meaning of the word prototype in this context is not quite that which it has in common engineering practice. The Guide makes clear that what is needed is a series of sets of computer programs not necessarily on the same hardware or using the same language as the operational software, which allow tests of operations and help to reveal operational problems as early as possible. The comments and recommendations of the working party are closely related to those of part 1 of the IECCA Guide.

The rules for funding and managing projects are quite reasonably drawn up in a way which is suited to engineering work. A software-based project is ill served by this framework, because the essence of software work is design. The project is only fully designed when the software is complete and working. Partly because of the nature of software, and partly because of current practice in writing it, a high risk exists at all earlier stages that some limitation has been overlooked. Our recommendations accordingly provide for contingency allowances in the initial estimates of hardware needs, and for the ability to expand the hardware (planned or actual) at later times. This may be needed when more accurate estimates can be made or when changes in circumstances call for new software. The recommendations are also couched in such terms as to encourage the production of working software at the earliest possible date.

The overall time for the development and operational use of a typical software-based project is long in comparison with the time for enormous improvements in the cost and power of computing hardware. This can lead to attempts to defer the choice of hardware to as late a date as possible in any given project. Problems which result from that are discussed in the next paragraph.

Software and hardware interact. Software cannot be tested without hardware, and hardware cannot be evaluated without software. Those projects which have depended on special-purpose hardware have not generally made good progress until the hardware was available with good facilities for program testing. The present rules for procuring software-based systems do not provide sufficient money for special-purpose hardware until there is evidence of significant progress. Deadlock is thereby achieved.

Much of the trouble we have noted concerns the problems which arise in assisting men to interact with a computer-based system and to interpret its information. This usually happens in a new and unfamiliar context, and there is no coherent body of theory on which to base decisions about the design of such interactions. In this situation there is a need for experiments which simulate real operating conditions as closely as possible. Projects have frequently lacked facilities for doing such simulations at a sufficiently early stage.

In certain cases there is experience of what is required in a context close to that of the system proposed. We have noticed that the procurement system is not so constructed that those designing the new system need to consult those operating the old one. Sometimes they do not do so.

The choice of manufacturer is, not unreasonably, governed to some degree by knowledge that his staff have experience in a certain field. The value of this is obvious, but there seems to be no system for ensuring that the results of experience are disseminated between the staff of manufacturers, research establishments, service establishments and the procurement executive. On the contrary, the procurement process may (in a NATO context) specifically exclude the manufacturer involved in project definition from the development contract. It also appears to be difficult to ensure that the staff which a manufacturer will employ on a given project will include those with the relevant experience.

There is a need for the procurement process to take account of the ability of software to 'stretch' to meet changing circumstances during the service life of a system. This should be one of the prime advantages of software-based systems, and it should lead to early thought about the most cost-effective way of ensuring a flow of ideas from users to those who can adapt the system to their needs. Although this process

should, and usually does, continue to the end of service life it can and should begin before handover and even before development starts. We have observed some welcome signs of adaptation of the procurement system to this need. The latest form of one particular Army command and control system appears to provide for repeated review. It is to be hoped that this will assist evolutionary development.

5 RECOMMENDATIONS

- (A) Software systems should be procured in such a way as to allow for continuous evolution all through their service life. This implies an approach to development by way of prototype systems (in the sense of the IECCA Guide) and also early arrangements for smooth handover of system responsibility to the user, who should be expected to provide in-service development.
- (B) The procurement system should provide for successive stages of project definition, each stage based on the lessons learnt from the prototype software produced at the previous stage. There is a need for an 'operational quality assurance' unit in each service to provide a focus of expertise for on-going development of operational systems. We believe that the group of 'rule writers' at ASWE (with broadened terms of reference) could well serve a useful purpose of this kind and that this form of organisation could with advantage be extended to other areas.
- (C) Encouragement should be given to the use of formal methods for investigating, and communicating, system design. Methods such as those of the Software Development System (SDS) (2) and the MASCOT system (3), developed by RSRE, clarify the functions and interconnections of modules of a system and provide a framework for effective management of the teams of software writers who will have to provide these modules.
- (D) Equipment for trying out the software of a system should be made available at the first possible stage in system design. This may be achieved in some cases by the use of a general purpose computer, possibly with special input and output devices. (See footnote). In other cases small (and increasingly economic) computers and peripherals may need to be purchased or borrowed from a pool. It is essential that whatever equipment is provided should give good facilities for program testing and should allow realistic simulation of the environment in which the system is expected to operate.
- (E) A software prototyping facility should be made available and much could be done by making use of the facilities which already exist at RSRE and ASWE. Attention should also be given to the possibility of constructing a portable facility which could be loaned to projects for relatively short periods.

FOOTNOTE: Experience using MASCOT within an evolutionary development for a current Army project has now proved that it is both possible and economic to transfer a software prototype from one type of computer to another. This approach enabled software development to be started significantly earlier than would otherwise have been possible. Based upon this experience a further project is planning the same approach (saving in this case almost one year of elapsed time).

(F) Procurement of computer hardware should take account of initial uncertainties concerning both the objectives to be attained and the size of the software for any given objective. For these reasons it should be accepted that it is cost-effective practice to carry adequate contingency allowances of store size, computing speed and general capacity in the hardware originally purchased, and to ensure that the hardware can later be expanded. Failure to do this is likely to lead to difficulty and expense in compressing the software to fit into the given hardware which could greatly exceed the cost of extending the hardware.

(G) The need for equipment to carry out software testing and evolutionary development during service should be recognised in considering hardware requirements. Due regard should be paid to the common need to test software thoroughly without interrupting a continuous service, and systems should carry hardware which can be 'insulated' from that providing service when testing is required.

(H) All reasonable means should be employed to increase contact at a technical level between those working for manufacturers and design authorities on the one hand and users on the other, so that the needs and experience of users can influence design and implementation at all stages. There should be free and frequent discussion when a project is under way. The working party are pleased to note that this was a recommendation of Cmnd 4641 (April 1971) para (12) page 14 (the 'Rayner Report'). It should be taken further, however, and it is very important that there should be encouragement for those working in the field of computer based projects to take part in more general discussion of problems which can lead to the formulation of projects.

6 REFERENCES

- 1 IECCA, A Guide to the Development of Computer Based Systems. IECCA (P) 4/72 Procurement Executive, Ministry of Defence.
- 2 F R Albrow, et al, The Pilot Software Development System, RRE Newsletter and Research Review No 14 1975.
- 3 K Jackson, H R Simpson, MASCOT - A Modular Approach to Software Construction Operation and Test. RRE Technical Note 778 October 1975.